Face detection without bells and whistles

Markus Mathias [1] Rodrigo Benenson [2] Marco Pedersoli [1] Luc Van Gool [1,3]

[1] ESAT-PSI/VISICS, iMinds
KU Leuven, Belgium

[2] MPI Informatics
Saarbrücken, Germany

[3] D-ITET/CVL
ETH Zürich, Switzerland

Abstract Face detection is a mature problem in computer vision. While diverse high performing face detectors have been proposed in the past, we present two surprising new top performance results. First, we show that a properly trained vanilla DPM reaches top performance, improving over commercial and research systems. Second, we show that a detector based on *rigid* templates - similar in structure to the Viola&Jones detector - can reach similar top performance on this task. Importantly, we discuss issues with existing evaluation benchmark and propose an improved procedure.

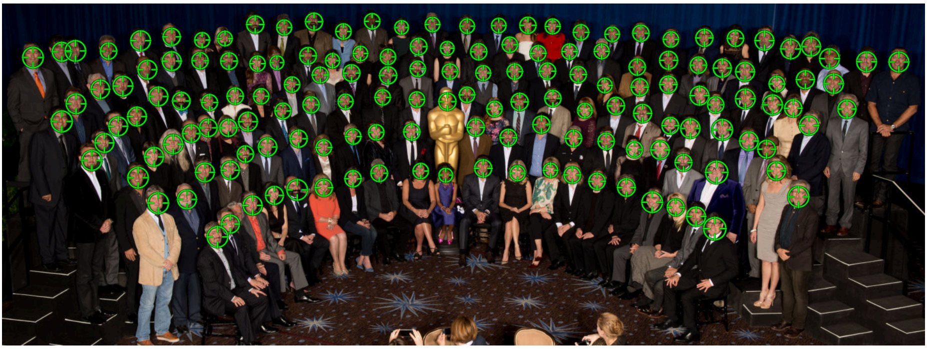


Figure 1: Our proposed `HeadHunter` detector at the Oscars. Can you spot the one false positive, and one false negatives ? (hint: first rows).

1 Introduction

Face detection is a classic topic in computer vision. It is a relevant problem due to its many commercial application in a human-centric world, and as a building block for more sophisticated systems. Deployed in a myriad of consumer products (e.g. digital cameras, social networks, and smart phone applications), it is considered a mature technology. In this paper we focus on the canonical problem of face detection in a single frame of photographs taken "in the wild".

Because of its maturity, we consider it as an application particularly suitable to study core aspects of object detection. One can expect benchmarking datasets with a diverse set of methods available for comparison. However, despite the interest in the topic and the quantity of data available, due to the lack of a commonly accepted annotation guidelines and evaluation protocols, a fair comparison of face detectors on various datasets is still missing.

In this paper we intend to create a common ground to evaluate and compare different face detectors. We have selected the most relevant datasets for face detection, improved their annotations, and propose a modified evaluation protocol that reduces dataset bias.

With this new evaluation in hand, we set to understand "what makes a face detector (truly) tick?". We propose to compare the well known deformable parts model (DPM) [9] with the integral channels detector approach [7]. We also compare side by side face detectors originating from the research community and from commercial products. We show that despite significant progress, face detection has not yet reached saturation. Even more surprisingly, we present new top results while using a simpler architecture than competitors. Although we focus on face detection, most of the discussion is agnostic to the object class.

1.1 Contributions

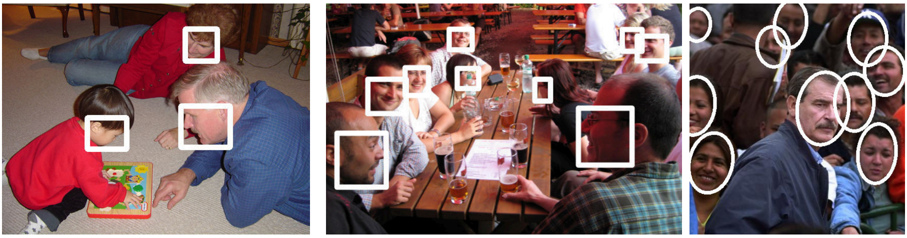
- We point out that the evaluation of existing face datasets is biased due to different guidelines for the annotation. We provide improved annotations and a new evaluation criteria that copes better with these problems (section 2).
- We show that (despite common belief) face detection has not saturated, and there are still relevant open questions to explore (section 6).
- We show that (contrary to previously reported results), when properly used, a vanilla deformable part models (DPM) [9] reaches top performance on face detection, improving over more sophisticated DPM variants (section 4).
- We evaluate for the first time an integral channels detector [7,3] for the task of face detection (section 3). We show that top detection results on face detection can be obtained using a small set of *rigid* templates (i.e. without deformable parts).
- We explore which aspects of such rigid detector most impact quality, such as the number of components or the training data volume (section 5).
- We provide source code for both our improved evaluation toolbox and for training/evaluating our proposed face detector.

1.2 Related work

Being a classic topic, there are probably thousands of papers specifically addressing the face detection problem. We present here a selection of what we consider landmark papers on the topic.

Nowadays the textbook version of a face detector is the Viola&Jones architecture [30]. It introduced the idea of computing an integral image over the greyscale input to enable fast evaluation of boosted weak classifiers based on Haar-like features. This detector provides high speed, but only moderate detection quality. This framework has been the source of inspiration for countless variants [35]. Amongst them SURF cascades [16] is one of the top performers (recently introduced by Intel labs).

Thanks to its elegant formulation, its intuitive interpretation, and strong results the Deformable Parts Model (DPM) has established itself as the de-facto



(a) Pascal Faces [33] (b) AFW [36] (c) FDDB [12]

Figure 2: Example frames of the annotated datasets considered.

standard for generic object detection [9]. This approach combines the estimation of latent variables for alignment and clustering at training time, the use of multiple components and deformable parts to handle intra-class variance, and a healthy dose of engineering to make it all work robustly and fast enough. A tree-structured DPM trained with supervised parts positions was successfully applied to face detection and fiducial points estimation [36,33], showing improved results over vanilla DPM.

Some of the earlier work on face detection employed neural networks [22,10,20]. Although competitive at the time, it is unclear how well such a method would perform on modern benchmarks. The work of [20] introduced the intriguing idea of coupling pose estimation and face detection into a single inference problem.

Other than the discriminative approaches mentioned above competitive results have been attained by formulating the detection problem as an image retrieval problem [27,17].

Instead of proposing a new detector, [13] shows that adapting a detector to the context of the test image can significantly improve detection quality. Although very interesting, it is a form of "per image semi-supervised learning". In this paper we focus on the raw detection problem, when using only the information available in each candidate detection window.

In our experimental section we also compare to black box commercial systems such as Picasa (from Google), Face.com (acquired by Facebook), Olaworks (acquired by Intel), and Face++ (start-up based in China).

2 Datasets

For our experiments we use four datasets of faces acquired in an unconstrained setup (so called "in the wild"). AFLW [15] contains $\sim 26\,000$ annotated faces, that we use for training. For preliminary experiments (sections 5.1 to 5.4), and parameters tuning we use the Pascal Faces dataset [33] (851 Pascal VOC images with bounding boxes). For comparison with previous work we use AFW [36] (205 images with bounding boxes) and FDDB [12] (2 845 images with ellipses annotations). See figure 2 for some example frames, which illustrate the "in the wild" aspect of our test data.

Unless otherwise specified detections are evaluated using the standard "intersection over union above 50 %" criterion [8], and quality is summarised using the average precision (AP).

2.1 Annotations and evaluation policies

The four datasets used in this paper are annotated by different research groups following different annotation strategies. As it stands, a face detector algorithm trained to output a specific bounding box policy cannot be properly evaluated directly on the different datasets.

In our preliminary experiments we found that adjusting the detector output towards the specific dataset annotations is key for competitive results. For most published methods it is unknown if adjustments to compensate different annotations have been made or not, making it difficult to perform a fair comparison. We want to improve this situation.

Differences in annotations Examples of dataset differences include: different policies for what constitutes a face (is a statue head a face? is a head rotated more than 90 degrees a face?), different sizes of annotation boxes (relative to the real world face, i.e. should the box span all facial landmarks, or include the whole head?), boxes centred differently (for lateral views, centred on the nose or on the cheeks?), and different minimum/maximum annotated face size.

All of these differences have a direct impact on the false positive and false negative evaluation metrics. If one method tunes for a specific dataset, then it will be unfairly penalized in another one. In this paper we take special care to design the comparisons as fairly as possible; we propose remedial measure for each of these issues. These measures require changes in the annotation and evaluation protocol for Pascal Faces and AFW (the FDDB dataset is immutable, see below).

New annotations The goal of new annotations is two fold: 1) Make sure that the bounding boxes are created using a uniform policy inside the dataset (this is imperative for proper evaluation). 2) To annotate all faces that might depend on the face presence policy.

For the new annotations, we adjusted the detection bounding boxes in Pascal Faces to match the guidelines defined in the supplementary material (similar to AFW one). The boxes in AFW already follow much stricter rules, and needed no major edits. Additionally, we added new annotations for overlooked faces and faces in challenging conditions such as small, occluded, or truncated faces. We labelled most of these new detections with an ignore flag. Methods should not be punished for their ability to detect challenging instances.

Remedial measures for bounding box policy Our new evaluation has a preprocessing stage that searches for a global rigid transformation of the detection output of a specific method, such as to maximize the overlap with the

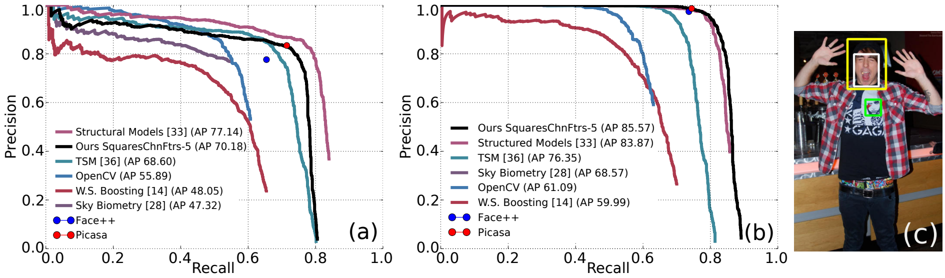


Figure 3: Precision-recall curves of the different evaluation methods on Pascal. (a) Shows the evaluation based on the previous annotations, not compensating different guidelines. (b) Transforms the detections to reflect the test set annotation policy. (c) Green and yellow boxes show different annotation/detection policies. The green box indicates a previously missing annotations, now marked as “ignore”. Detecting this face should not count as false positive.

ground truth annotations. By searching a global scaling and translation that maximize performances we evaluate as if each method would have taken care of targeting their detections (size and position) towards the specific test set.

Note that since bounding boxes are adapted for every method in our evaluation, it becomes part of the evaluation protocol and does not advantage any specific method. The details of the estimation algorithm are provided in the supplementary material.

Remedial measures for different scale ranges Another important aspect of the different detectors is their minimal and maximal search scale. Different search ranges result in different sets of detected bounding boxes. The search range and annotation quality/guidelines have severe impact on the overall detector quality. If one approach searches for smaller faces than specified by the dataset policy, high scoring false positives might be introduced; if a method is searching only for larger faces, it will miss out on recall. Thus using annotations and detections as-is is a no go.

For the sake of explanation let us assume a dataset has been perfectly annotated for all faces larger than 15 pixels. Different detectors will output different detection sizes, which might or might not cover the minimum size annotations. In this example, let us assume that we are interested in evaluating all faces larger than $\alpha = 30$ pixels. The naive approach would be to chop-off all annotations smaller than α, and also all detections smaller than α. However, if the detector originally triggered with a bounding box of size $\alpha - 1$ for a face of true size α, removing it will create a drop in recall (false negative). If one decides to keep detection smaller than α while dropping annotations smaller than α, then this create artificial false positives. The naive approach does not work either.

We propose to solve this problem in the following way. Given a set of annotations, the evaluation minimal size α is set to a value comfortably larger than the minimal annotation size. We introduce a second threshold β, which defines the minimal size of *detections* that we consider. We set $\beta = \sqrt{0.5 \cdot \alpha^2}$, given that our overlap over union threshold is 0.5. With β we keep all detections which would still have sufficient overlap (> 0.5 overlap over union) with a ground truth bounding box of size α, and remove all smaller ones. Finally, to avoid small false positives, we mark all annotations smaller than α with the "ignore" flag. With these two thresholds we reduce the border effects, and obtain the desired unbiased evaluation. In our evaluation we set set α to 30 pixels.

Impact of the new protocol To summarize our new protocol for Pascal and AFW datasets include: a) new annotations, b) a transformation of the detection bounding boxes to adapt each algorithm to each dataset, c) a new handling of detection windows on the border of the annotated scale range.
To give an impression about the importance of a proper evaluation, in figure 3 we compare the precision-recall curves of several methods on the Pascal Faces dataset. Sub-figures 3a and 3b show, respectively, results with the original annotations and the standard protocol (Pascal VOC [8]), and with our new annotations and protocol. Many detections, which are counted as errors in figure 3a are actually wrongly annotated. This produces an artificial slope on all the curves that biases the results. Importantly, notice how the change of evaluation protocol (from figure 3a to figure 3b) also produces a *different ranking* for the methods.

FDDB dataset This dataset has a good annotation quality, provides a publicly available evaluation toolbox, and collects results online. All of these are best practices. Unfortunately, the FDDB protocol calls for sharing the ROC curves, not the detection bounding boxes. Without these boxes it is impossible to improve the evaluation, or to have a in-depth analysis of the different detection methods. We do not (cannot) use our new evaluation protocol for the FDDB dataset.
For our own methods we convert our detection bounding boxes into ellipses based on the dataset annotation description [12]. The FDDB evaluation protocol foresees to match bounding boxes with their annotation ellipses using the Pascal VOC criterion. Changing the output format from bounding boxes to ellipses immediately increases the overlap region, showing a significant positive impact on the result curve. Here again, it is unclear which other methods make similar adjustments.

Our evaluation tools, and the new annotations for Pascal Faces and AFW, will be released together with this paper. We hope that future detection benchmarks will consider in their design the issues raised here.

3 Integral channel features detector

One of the key ingredients in the classic Viola&Jones face detector [30] is the use of an integral image (summed area table) for fast features computation. This idea is generalised by the integral channels features framework described in [7]. Instead of computing an integral image over a single input greyscale channel, it is proposed to define an arbitrary set of feature channels (feature maps), such as quantised oriented image gradient, colour, linear filter responses, etc. The integral images defined over these channels allow to quickly sum over arbitrary rectangular pooling regions. The object detector is trained by boosting a forest of trees built using such rectangular pooling regions as input features.

Somewhat surprisingly, this combination of classic ingredients (oriented gradient and colour feature maps, decision trees, and Adaboost) has shown top performance on tasks such as pedestrian detection [3], traffic signs detection [19], and feature points matching [29]. It reaches higher pedestrian detection quality than more sophisticated methods using deformable parts [9], more complex features [31], non-linear kernels [18] or a deep architecture [26].

We propose to adapt the integral channels detector to the task of face detection. We purposely use a plain setup, similar to [7,3,19]. Unless otherwise specified we use simple gradient magnitude channels (six for quantised orientations, one for magnitude channel), and colour channels (LUV colour space). We use shallow boosted trees of depth two (three stumps per tree).

The main difference from previous instances of this framework is that instead of using a single template per object category, we combine a set of templates to represent the face category (so called "components") [9,25]. Each component captures a fraction of the intra-class diversity of faces. At test time all templates are evaluated, and their detections merged during non-maximum suppression.

3.1 Baseline detector

Our baseline detector `SquaresChnFtrs-5` consists of 5 components, clustered using the yaw angle annotations. We collected a total of 15 106 samples from the AFLW database [15] to train 5 models (components) of size 80 × 80 pixels.

A frontal face detector (yaw angle ±20 degrees) and two side views (20 → 60 and 60 → 100 degrees) are trained using 6 752, 5 810, and 2 544 samples respectively. Pitch and roll are kept between ±22.5 degrees. As negative samples we use 3 652 person-free images from the Pascal VOC database [8]. The remaining two models are mirrored version of the side views. See supplementary material for details on the learned models and their training samples.

For each component the training is similar to the `SquaresChnFtrs` setup described in [3], unless otherwise specified we use the same parameters. The features are drawn from a pool containing all possible square features (28 700). We perform 4 rounds of bootstrapping to ensure that no additional false positives can be found in our negative training data. Our final component detector consists of an Adaboost classifier with 2 000 weak learners. For non-maximum suppression we join the candidate detection from all components and suppress them together using the overlap over min-area criterion as described in [7, addendum], the overlap threshold is set to 0.3.

In the experiments of section 5 we explore the design space of our detector, and in section 5.5 we describe a stronger detector than our baseline.

3.2 Detection speed

By using aggressive (soft and crosstalk) cascades and reducing features computation across scales [2,6], it has been shown that integral channels detectors can reach fast detection rates of the order of ~ 50 Hz for 640×480 pixels images (either on GPU or multi-core CPU). The bulk of time is spent in the weak classifiers evaluation.

In our setup adding more templates to evaluate costs a linear decrease in speed. This could be mitigated by using a hard cascade where a short classifier is first evaluated (trained on all views), before deciding to evaluate all view-specific classifiers. Our implementation is not speed-aware, however even with a conservative estimate (5 components → 5 × slow-down) it seems reasonable to believe that the proposed approach can reach frame-rate detection speeds of ~ 10 Hz once tuned for speed.

4 Building the DPM baseline

Other than considering an evolved version of the Viola&Jones detector, we would like to also consider an evolved version of the classic HOG+SVM [4]. As a reference baseline we train a DPM using the same training data as SquaresChnFtrs-5. We use publicly available DPM version 5 [9].

We define the DPM components using the same three views as SquaresChnFtrs-5 (defined in section 3.1), due to mirroring this results in 6 components. Each component has one root template and 8 parts. Besides the initialization of the components we keep all other training parameters to default.

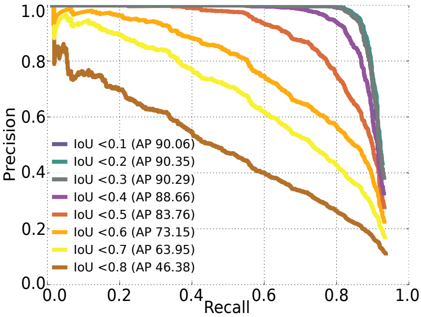


Figure 4: For the DPM detector, the non-maximum suppression overlap (intersection over union, IoU) threshold is an important parameter. The default value of 0.5 leads to poor performance.

We found that a test time the non-maximum suppression (NMS) overlap threshold is a crucial parameter. Figure 4 shows our DPM evaluated over the Pascal Faces dataset using different thresholds. When using the default value 0.5 the detection performance is significantly lower than when using an an adequate one (we use 0.3). When setting the NMS threshold to the default value of 0.5, the low performance DPM results are consistent with the previously reported one [36]. In the supplementary material we have the equivalent plot for SquaresChnFtrs-5.

As will be discussed in section 6, to our surprise, our DPM baseline turns out to match or outperform *all other methods,* including more sophisticated DPM variants. We attribute the strong results to the proper use of available training data, and to noticing the importance of appropriate non-maximum suppression.

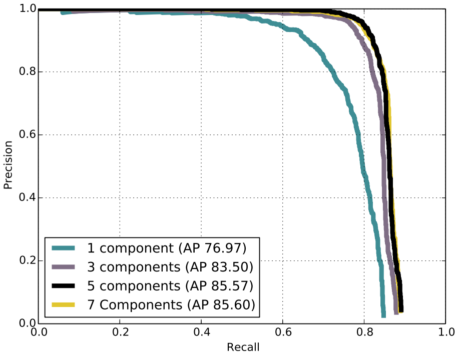


Figure 5: Quality versus number of components. AP: average precision.

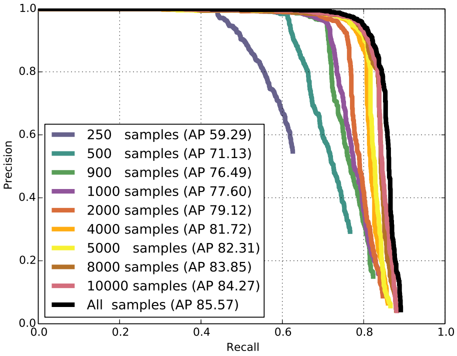


Figure 6: Quality versus number of training samples.

5 Experiments

We are interested in understanding which ingredients are critical for good face detection. The effect of the parameters of the integral channels detector have been explored in previous work on pedestrian detection [7,3]. A few of them are repeated for faces and reported in the supplementary material (overall we observe the same trends). In this section we present experiments (over the Pascal Faces dataset) that explore orthogonal aspects not covered by previous work, regarding view-specific components (§5.1), the amount of training data (§6), and skin aware feature channels (§5.3). Along them, when relevant, we draw parallel comparisons with the DPM approach. In all plots, the thick black line corresponds to our SquaresChnFtrs-5 baseline detector (§3.1). In section 6 we provide a comparative evaluation with other face detectors.

5.1 How many multi-view components ?

The number of components is considered to be a critical ingredient for high quality detections [5]. Figure 5 shows the impact of the number of components on the detection quality. When adding new components we only change the steps of the yaw angle (instead of introducing views which where not considered in our baseline model, such as faces with $>$ 22.5 degree roll and pitch angles).

It can be seen that the quality of the integral channel features detector does not improve any further past 5 components. As an increase in the number of components directly maps to a decrease in detection speed, using more components seems not to be beneficial for our use case. Choosing 5 components for our baseline strikes a good balance between detection quality and detection speed. If accurate face pose detection is of interest, more components may help to get better initial pose estimates.

Our comparative experiments with DPM are done using 6 components, these are the same 5 components, plus one obtained by mirroring the frontal face (default behaviour of the DPMv5 source code [9]).

5.2 How much training data ?

Collecting training data is a labour intensive task. The different methods evaluated in this paper differ in the number of training samples (900 to > 20 000) and also in the type of annotations (from simple bounding boxes to facial landmarks and face orientation). The amount and quality of the training data can highly influence the performance of a detector. Exploring the influence of the amount of annotations on all other methods is beyond the scope of this paper, we have to assume that other methods explored this option to present a competitive face detector.

To quantify how our approach scales with the amount of training data, we evaluate the impact of varying the amount of training data in terms of precision and recall. In figure 6 we plot the precision-recall curves on the Pascal Faces dataset when gradually varying the number of samples from 250 to the entire training data. Our `SquaresChnFtrs-5` performs quite poorly when trained with only a few samples. By adding more training data the recall can be steadily improved without affecting precision. Our results indicate that the detection quality could further be improved by using an even larger training set.

When doing a similar experiment for our `DPM` we observe that with as few as 500 samples it reaches already 95 % of its final average precision (AP) value. Similar to `SquaresChnFtrs-5`, increasing the number of samples improves its recall.

The number of training samples also highly influences the training time. When using all available training data, `SquaresChnFtrs-5` will be trained in less than 6h, while DPM needs roughly one week.

5.3 Which colour channels ?

One difference between our baseline detector (§3.1) and a vanilla `HOG` template (used in `DPMs`), is the use of `LUV` colour channels. Since faces have a discriminative colour distribution, one wonders how much colour helps for the task. In figure 8 we investigate the effect of colour for face detection. We consider the following channels (see figure 7): `HOG`, the gradient magnitude and quantized orientations; the `L` luminance channel (grey image); the `U` chromaticity channel, which is known to respond to skin colour; `RmG` is the subtraction of the red and green channels, included because $20 < \texttt{R} - \texttt{G} < 80$ is the simplest known skin colour detector [1]; `Skin` is a naive Bayes skin colour classifier trained on the dbskin dataset [23].

The results figure 8 shows that the colour information mainly affects the recall. Unsurprisingly the `Skin` channel is the most informative, we also confirm that `U` captures relevant information for skin detection, improving over `RmG`. Even the weakest colour channel improves over the `L` greyscale channel, indicating that chromaticity is indeed relevant for the task.

Finally, when probing combinations such as `HOG+L+Skin` or `HOG+LUV+Skin` we see no improvement compared to the vanilla `HOG+LUV`. This indicates that, for this task, the classifier is able to extract the relevant information directly from the `LUV` channels, without requiring the use of custom made channels.

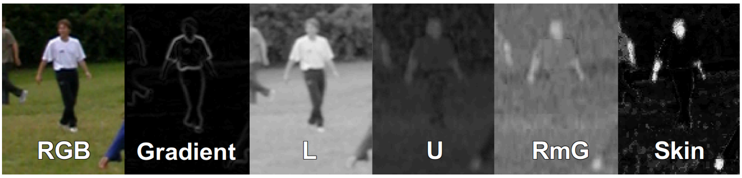


Figure 7: Example of colour channels considered, see section 5.3.

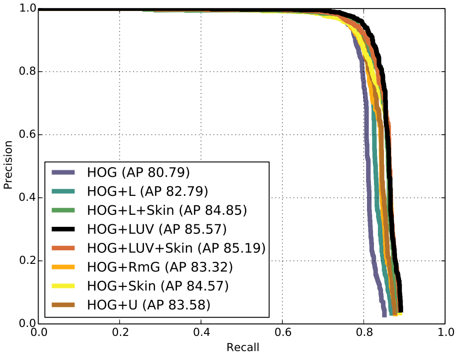


Figure 8: Quality versus colour channels.

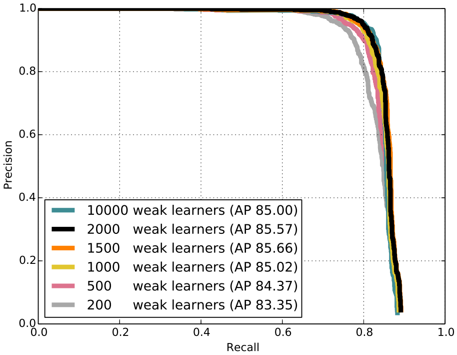


Figure 9: Quality versus number of weak classifiers.

5.4 How many weak classifiers ?

The number of weak classifiers boosted to build the strong classifier is an important parameter which is usually set to a fixed value. We observed that during training, already a small amount of weak learners is enough to successfully separate the positive and negative samples (20 stages before bootstrapping, 100 stages after the last bootstrapping stage). Since Adaboost lacks a well understood regularization mechanism [24], depending on the training data, adding more weak classifiers could lead to over-fitting.

Figure 9 shows the influence of the classifier length on the detection quality. A small amount of only 200 weak learners is already enough to get decent detection quality of 82.8% average precision. Since weak classifiers evaluation is the speed bottleneck, using a smaller number of weak learners is of special interest when targeting high detection speed.

On the other side, it can be observed that even with 10 000 Adaboost stages, the performance does still not deteriorate. This shows that when using faces for training, the system is robust to the number of weak classifiers.

To match previous setups [7,3] we set the number of weak classifiers in our baseline to 2000, even though figure 9 shows that the detection quality already saturates with 1500 classifiers.

5.5 Building a strong face detector

For our final face detector model we focus on quality to see how competitive a detector based on rigid templates can be. To that end we apply previous results presented in [3]. In that paper, the authors present three strategies to improve the quality of an integral channel features detector. First, we follow their

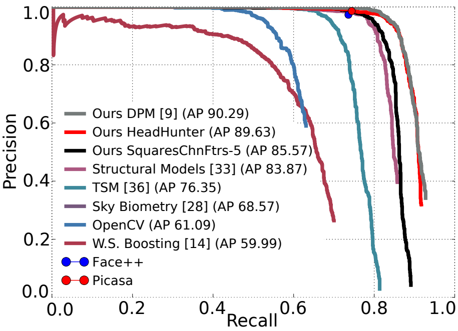


Figure 10: Pascal Faces results.

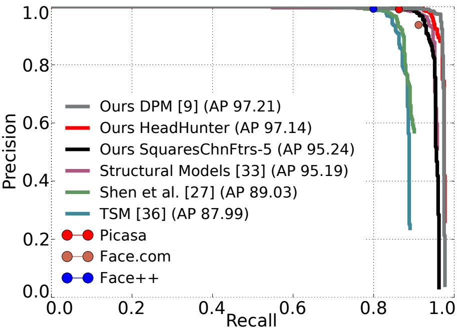


Figure 11: AFW results.

suggestion by applying global normalization [21] before running the detector. Second, we train a multi-scale model by doubling each template (component) with an additional one of twice its size. Third, the templates are trained using the maximum amount of pooling features: all possible rectangles for the the baseline, and all possible squares for the largest templates (see [3]).

As can be seen by the high average precision of our baseline (e.g. figure 5), most views are well captured by our training data. On the other hand, as mentioned in section 5.2, being a rigid model, our detector has difficulties to handle unseen views (compared to a DPM, which generalizes via deformations). To improve recall we add copies of the training data with a rotation of 35 degrees. We use these to train 6 additional components that handle tilted faces. Using the eleven $(5 + 6)$ components together provides further improvement in detection quality (mainly increase in recall).

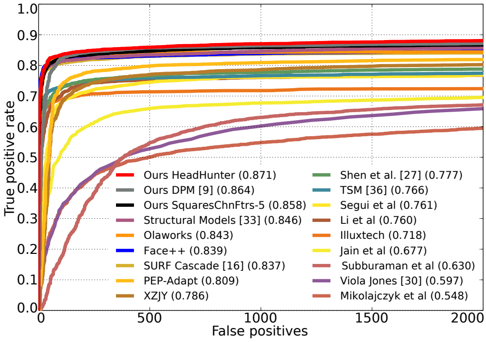


Figure 12: FDDB results.

We name our final strong multi-scales model the `HeadHunter` detector. This detector consists in total of 22 templates, 11 for each scale. Each scale uses 5 templates for the frontal faces and 6 for the rotated faces. We train a total of 12 different templates, the remaining 10 templates are generated via mirroring.

6 Comparative detection quality

Figures 10, 11, and 12 show the results of our methods compared to many competitors (including research and commercial systems). Only a few methods provide results on all three datasets.

Commercial systems The commercial systems often do not provide confidence score and are shown as a single point. As can be seen these methods are among the best performing ones with an operating point clearly chosen to provide high precision.

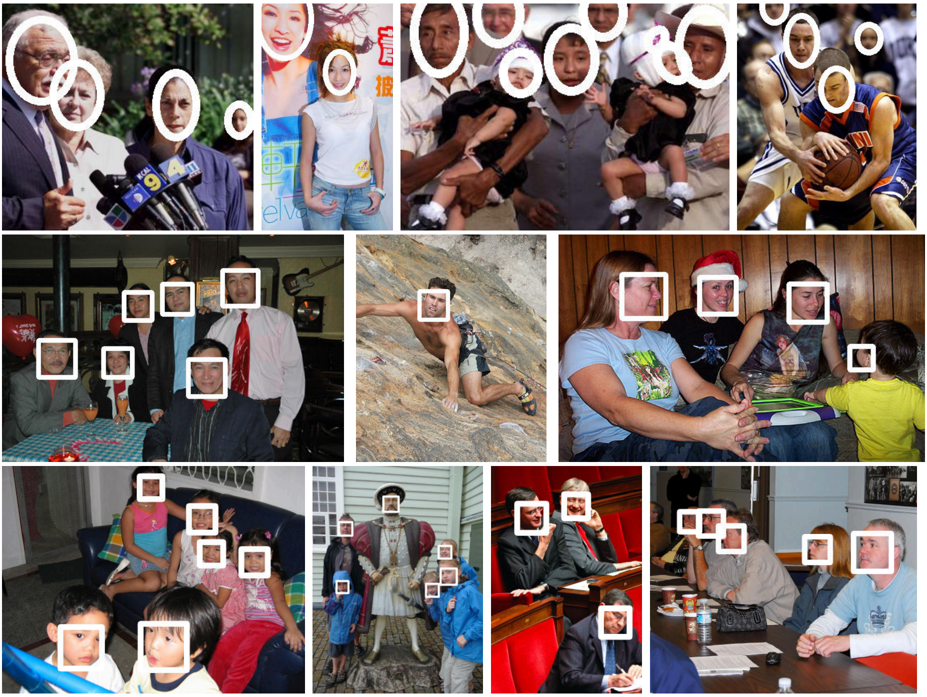


Figure 13: Qualitative HeadHunter detection results from FDDB (top row), Pascal Faces (middle row), and AFW (bottom row).

Annotation type One of the highest scoring competitors to ours is the DPM based structured models method [34,33]. However, similarly to TSM (Tree Parts Model) [36], this method requires the annotation of facial landmarks, while we need only the bounding boxes. Furthermore, the method uses also context (upper body detector) to improve results, although it is not clear how important that is for the final results. We note that already a single template of our baseline (figure 5) matches the performances of TSM [36].

Common approaches Most approaches rely either on a Viola&Jones like framework (e.g. Face++,), or HOG+SVM based (e.g. TSM, Structured Models, DPM). Even if methods are based on these two frameworks the range of results can vary a lot. This underlines once again how the task of object detection is sensitive to small details and therefore in depth analysis such as this one are required.

DPM results Overall it is quite striking to notice that a properly trained DPM baseline obtains top performance across all datasets considered (updating previously reported results, such as [36]). This is a testament to the importance of careful baselines design, the importance of low-level details (a single threshold value makes the difference between under-performing to top performing), and the value of open source release of research material.
In parallel to the preparation of this manuscript Yan et al. [32] independently reported results DPM for the AFW dataset which are consistent to our results.

Their work however is not focused on detection quality, and their high performing results are left unexplained there. Our discussion of section 4 details the critical ingredients for a high quality DPM face detector.

Rigid templates Although our DPM reaches top performance, the experiments also show that HeadHunter, a set of rigid templates, essentially reaches the same performance. This indicates that parts are useful but not critical to reach top performance. As long as enough training data is available to cover pose diversity, a small set of rigid templates will detect faces as good as anything else.

Problem saturation The difference in recall at high precision between 11 and 10 indicates that when increasing dataset difficulty, existing methods fail to reach full recall. This shows there is still a measurable gap before matching human performance. The missing recall in 12 seems mainly due to out-of-focus image blur, one could consider this a separate problem. A detailed analysis of the causes of failure for each detector type still remains to be done [11].

Open questions Not only detection quality has not saturated, but also multiple questions remain open, for example: the DPM and HeadHunter use mainly orthogonal strategies to improve detection quality; how can deformable parts and strong boosted templates be used together best? If blur causes missing recall in FDDB; how to best handle this case? There is not yet strong evidence that fiducial points annotation can help build better detectors; how best to exploit this data?

7 Conclusion and future work

In this work we have shown that even if face detection is a quite mature field, there is still room for improvements in terms of both detection performance as well as evaluation protocols. We have shown that the evaluation protocol plays an important role, analysed the current issue and provided a thorough and fair evaluation of face detectors in different datasets. We also provide a update evaluation method, which might well be suitable for other detection evaluation datasets.

It turns out that for face detection the children of two classic detection approaches, Viola&Jones and HOG+SVM, are the best performing methods. Both our DPM and integral channel features model, HeadHunter, reach top performance on the task. Rigid templates provide excellent quality for many classes, especially if sufficient training data is available. DPMs are still the method of choice if only few training samples are available and at the same time high recall is of essence. We believe that our findings are an important cue for the next generation of detectors, probably combining the capacity of representation provided the integral channel features detector with the powerful generalization induced by modelling deformations.

Acknowledgement Many thanks to Junjie Yan for access to Pascal Faces result curves. Work supported by the EU project EUROPA2, and the Cametron project.

References

1. Al-Shehri, S.A.: A simple and novel method for skin detection and face locating and tracking. In: CHI. Springer (2004)
2. Benenson, R., Mathias, M., Timofte, R., Van Gool, L.: Pedestrian detection at 100 frames per second. In: CVPR (2012)
3. Benenson, R., Mathias, M., Tuytelaars, T., Van Gool, L.: Seeking the strongest rigid detector. In: CVPR (2013)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
5. Divvala, S., Efros, A., Hebert, M.: How important are deformable parts in the deformable parts model? In: ECCV, Parts and Attributes Workshop (2012)
6. Dollár, P., Appel, R., Kienzle, W.: Crosstalk cascades for frame-rate pedestrian detection. In: ECCV (2012)
7. Dollár, P., Tu, Z., Perona, P., Belongie, S.: Integral channel features. In: BMVC (2009)
8. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge (2008)
9. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. PAMI (2010)
10. Garcia, C., Delakis, M.: Convolutional face finder: A neural architecture for fast and robust face detection. PAMI (2004)
11. Hoiem, D., Chodpathumwan, Y., Dai, Q.: Diagnosing error in object detectors. In: ECCV (2012)
12. Jain, V., Learned-Miller, E.: Fddb: A benchmark for face detection in unconstrained settings. Tech. Rep. UM-CS-2010-009, University of Massachusetts, Amherst (2010)
13. Jain, V., Learned-Miller, E.: Online domain adaptation of a pre-trained cascade of classifiers. In: CVPR (2011)
14. Kalal, Z., Matas, J., Mikolajczyk, K.: Weighted sampling for large-scale boosting. In: Everingham, M., Needham, C.J., Fraile, R. (eds.) BMVC. British Machine Vision Association (2008)
15. Koestinger, M., Wohlhart, P., Roth, P.M., Bischof, H.: Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In: ICCV BeFIT workshop (2011)
16. Li, J., Zhang, Y.: Learning surf cascade for fast and accurate object detection. In: CVPR (2013)
17. Ma, K., Ben-Arie, J.: Vector array based multi-view face detection with compound exemplars. In: CVPR (2012)
18. Maji, S., Berg, A., Malik, J.: Classification using intersection kernel support vector machines is efficient. In: CVPR (2008)
19. Mathias, M., Timofte, R., Benenson, R., Van Gool, L.: Traffic sign recognition - how far are we from the solution? In: ICJNN (2013)
20. Osadchy, M., LeCun, Y., Miller, M.: Synergistic face detection and pose estimation with energy-based models. JMLR (2007)
21. Rizzi, A., Gatta, C., Marini, D.: A new algorithm for unsupervised global and local color correction. Pattern Recognition Letters (2003)
22. Rowley, H., Baluja, S., Kanade, T.: Neural network-based face detection. PAMI (1998)

23. Ruiz-del-Solar, J., Verschae, R.: Skin detection using neighborhood information. In: FG. pp. 463–468. Seoul, Korea (May 17-19 2004)
24. Schapire, R.E.: Explaining adaboost. In: Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik (2013)
25. Schneiderman, H., Kanade, T.: Object detection using the statistics of parts. IJCV (2004)
26. Sermanet, P., Kavukcuoglu, K., Chintala, S., LeCun, Y.: Pedestrian detection with unsupervised multi-stage feature learning. In: CVPR (2013)
27. Shen, X., Lin, Z., Brandt, J., Wuk, Y.: Detecting and aligning faces by image retrieval. In: CVPR (2013)
28. SkyBiometry: www.skybiometry.com
29. Trzcinski, T., Christoudias, C.M., Fua, P., Lepetit, V.: Boosting binary keypoint descriptors. In: CVPR (2013)
30. Viola, P., Jones, M.: Robust real-time face detection. In: IJCV (2004)
31. Walk, S., Majer, N., Schindler, K., Schiele, B.: New features and insights for pedestrian detection. In: CVPR (2010)
32. Yan, J., Lei, Z., Wen, L., Li, S.Z.: The fastest deformable part model for object detection. In: CVPR (June 2014)
33. Yan, J., Zhang, X., Lei, Z., Li, S.: Face detection by structural models. Image and Vision Computing (2013)
34. Yan, J., Zhang, X., Lei, Z., Li, S.: Real-time high performance deformable model for face detection in the wild. In: ICB (2013)
35. Zhang, C., Zhang, Z.: A survey of recent advances in face detection. Tech. rep., Microsoft Research (2010)
36. Zhu, X., Ramanan, D.: Face detection, pose estimation, and landmark localization in the wild. In: CVPR (2012)

Face detection without bells and whistles Supplementary material

Markus Mathias [1] Rodrigo Benenson [2] Marco Pedersoli [1] Luc Van Gool [1,3]

[1] ESAT-PSI/VISICS, iMinds
KU Leuven, Belgium

[2] MPI Informatics
Saarbrücken, Germany

[3] D-ITET/CVL
ETH Zürich, Switzerland

1 Non-maximum suppression

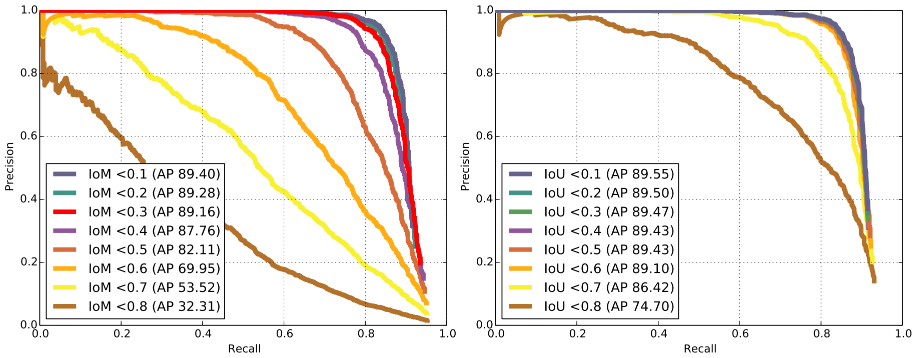
In the main paper we already show the importance of the non-maximum suppression to receive competitive results with DPMs. Here, we summarize the impact of the non-maximum suppression parameters and also show its effect on the `HeadHunter` detector. The overlap threshold and how the overlap is computed both play an important role when performing greedy non-maximum suppression.

The goal of non-maximum suppression is to keep only one detection per object instance, selecting the highest scoring detection and automatically removing all redundant (lower scoring) detection bounding boxes that refer to the same object. If two objects of interest are in close distance, e.g. occluding each other, overlapping boxes should be kept if they refer to different objects.

Given two candidate detection bounding boxes, the PASCAL VOC [1] overlap criterion is defined as the intersecting area divided by their union (so `IoU` criterion). For pedestrian detection Dollar et al. [2, addendum] introduced a different criterion defined as intersecting area divided by the area of the smaller box ("intersection over min-area" or `IoM`). Figure 1 compares the effect on detection quality of the overlap measure and its threshold when doing non-maximum suppression. The curves reflect the performance of the `HeadHunter` detector on the Pascal Faces dataset. To generate these curves we performed an initial detection run with a (`IoU` or `IoM`) overlap threshold of 0.8 (which retains many false positives), the curves for other overlap thresholds are then generated via post-processing. The shown curves are therefore a close approximation to the results obtained when evaluating the full pipeline with each different overlap threshold.

Since the Pascal Faces dataset does not contain many face to face occlusions, the overlap threshold plays a minor role when chosen lower than a certain threshold (< 0.5 in case of `IoU` and < 0.3 for `IoM` criterion). On the other hand a different dataset may have many true positives located next to each other in the image, prohibiting to simply choosing 0.0 as overlap threshold.

To maximize recall our experiments suggest to select a threshold close to the 0.4 for the `IoU` criterion and 0.3 for `IoM`'s. Our final `HeadHunter` detector uses `IoM` set to threshold 0.3.



(a) `IoU` intersection over union criterion (b) `IoM` intersection over min-area criterion

Figure 1: Influence of the overlap thresholds for different non-maximum suppression criteria. `HeadHunter` detector evaluated over the Pascal Faces dataset.

2 Annotation corrections

Even tough we minimize the influence of different annotation guidelines between training and testing to provide a better evaluation method, the quality of the underlying annotations still plays an important role when comparing different face detection methods.

As discussed in section 2.1 of the main paper, to improve the quality of evaluation we revisited the AFW and Pascal Faces datasets and we added bounding boxes for faces previously not annotated. For Pascal Faces we also adjusted the annotation bounding boxes of existing annotations, since these did not follow a clear annotation policy. Some of the issues found in these datasets are visualized in figures 2 and 3.

Please note that ambiguous faces are marked as "ignore region", and that very small and very large annotations are better handled via our improved evaluation protocol (see main paper).

3 Annotation policy

We annotated all faces with one side longer than 15 pixels. The bounding boxes are tight so as to include all 21 facial landmarks defined in [3] and illustrated in figure 4.

We annotated occluded faces if at least on eye is visible in the image. The bounding box is tight around the visible facial landmarks, we do not hallucinate the non-visible area. Bounding box annotations also do not span over image borders.

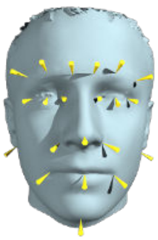
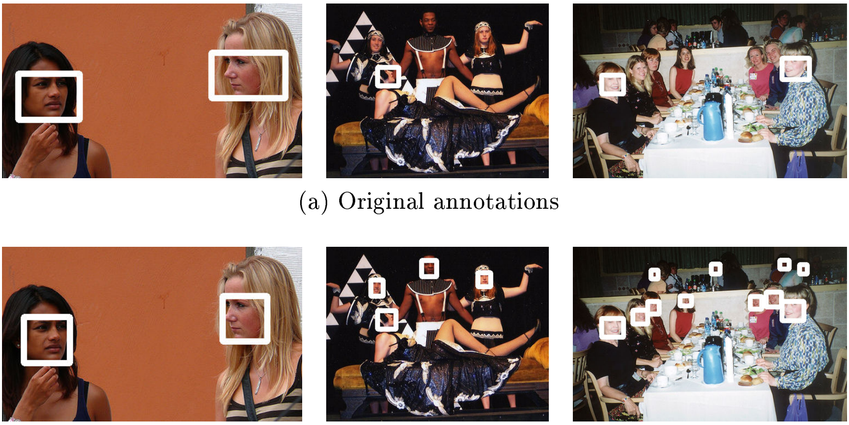


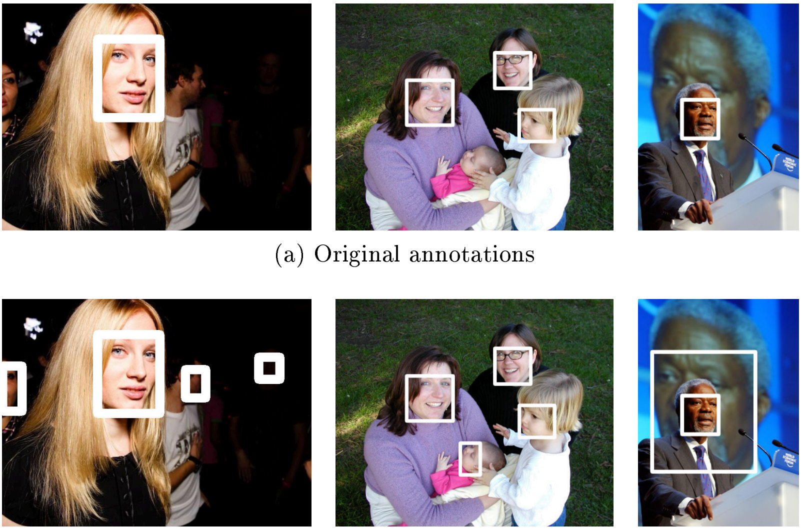
Figure 4: Bounding boxes include all indicated facial landmarks. Image from [3].



(a) Original annotations

(b) Updated annotations

Figure 2: Original and updated annotations on the Pascal Faces dataset. We added many missing bounding boxes and changed existing ones to follow a consistent annotation policy.



(a) Original annotations

(b) Updated annotations

Figure 3: Original and updated annotations on the AFW dataset. We only added missing annotations, the bounding boxes of existing annotations were already accurate.

Except for clear annotation mistakes, we marked new annotations with the "ignore" flag. Although many of these additional annotations show faces in difficult condition, we did not want to change the overall characteristic of the datasets. By adding annotations for faces which might be difficult to detect, we do not punish algorithms which are able to detect such instances, especially if they do so with high confidence. We found that in the original Pascal Faces dataset 19 % of faces were not annotated.

4 Remedial measures for bounding box policy

For a fair comparison among methods with different bounding box annotation policy, we optimize the size and location of the detection bounding boxes for each method independently with the goal to maximize the overlap with the ground truth annotations (see section 2 of main paper). By searching a global scaling and translation that maximize performances we evaluate as if each method would have taken care of tuning their detections towards the specific test set. Note that since bounding boxes are adapted for every method in our evaluation, it becomes part of the evaluation protocol and does not advantage any specific method.

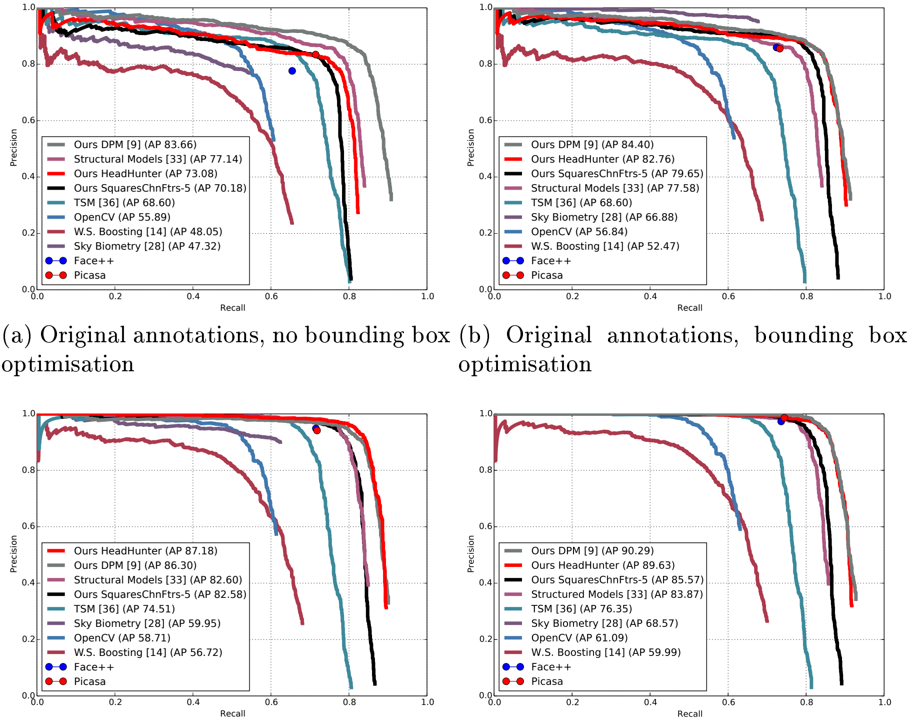
For estimating this per-method transformation we collect the mean displacement in term of location and size of all correct detections with the corresponding ground truth annotations (fulfilling the required intersection over union > 0.5). These values represent the bias between annotations and detection bounding boxes. We use them to estimate a global transformation that attempts to correct the bounding box size, ratio, and location. This way we obtain detections that are better aligned with the annotations. Since the parameters we optimize influence each other, we iteratively repeat this procedure several (five) times.

Figure 5 shows the effect of correcting the annotations and removing the mentioned bias.

4.1 Single method, multiple policies

In the case of detections from Google Picasa, we realised that the bounding boxes have a very high variation in size for different viewing angles, such that frontal faces span the whole head, while lateral boxes are tight, similar to our annotation guidelines. Such effects could be caused (internally) by using different detectors with different bounding boxes policies. The procedure described in the previous section works properly only when there is a single mode in the location and size biases. With multiple modes, as observed for the Picasa bounding boxes, the method would compensate only for the mean error.

In this particular case we manually verified all detections. Other methods seem not to have such severe variations. Adjusting boxes differently for each detection mode would be preferable but requires additional information currently not provided by most methods.



(a) Original annotations, no bounding box optimisation

(b) Original annotations, bounding box optimisation

(c) New annotations, no bounding box optimisation

(d) New annotations, bounding box optimisation

Figure 5: Precision/Recall curves of the different evaluation methods on the Pascal Faces dataset. (a) Shows the evaluation based on the original annotations, not compensating different guidelines. (d) Shows our new evaluation. (b) and (c) are partial combinations.

5 Learned `SquaresChnFtrs-5` model

The learned `SquaresChnFtrs-5` model has multiple aspects to it, which we visualize in the following series of figures. Figure 6 shows the three learned components (the two side view models are mirrored, making a total of 5 templates at run-time). Figure 7 shows the spatial distribution of the pooling regions per channel. Different from DPM model, the integral channel features model has overlapping pooling regions and thus no clear pattern emerges.

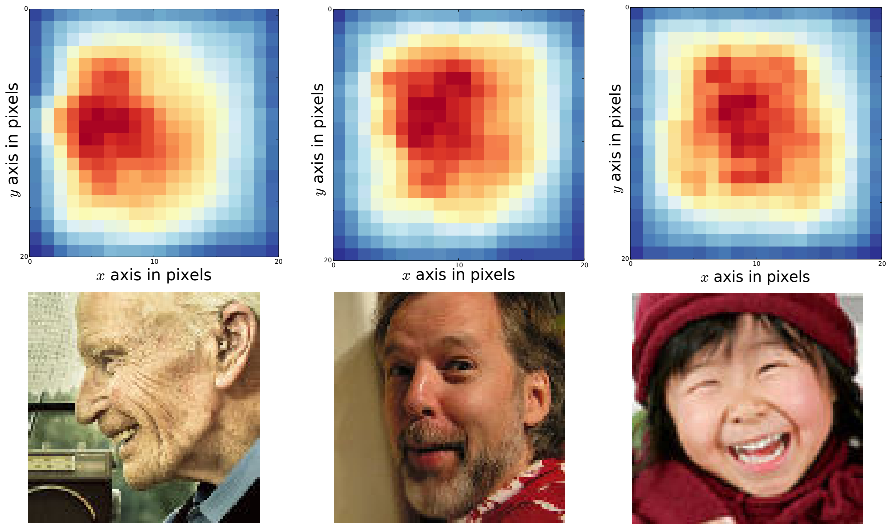
Figure 8a shows which channels are more commonly selected, and figure 8b shows which pair of channels are more commonly used (in each branch of the level-2 decision trees). It can be seen that most decisions are based on the luminance channel, horizontal gradients, and gradient magnitude channels. Interestingly, it can be seen that the luminance channel is mainly used with itself, while other channels tend to mix with each other (as would be expected). The side view models (not shown) have a similar pattern for the channels usage.

A different view of the spatial arrangement of the pooling regions is visible in figure 9. Figure 6c shows the learned frontal face template, figure 7c presents it decomposed by channel, and figure 9a does so by decomposing by pooling region area. In figure 9a it can be seen that most pooling regions are small, the most relevant ones (i.e. higher detection score influence) covering the face area. Only a few features cover the whole template.

Figure 9b shows the centre of each (square) pooling region, and figure 9c the pairs of features used in each level-2 decision tree branch (one line segment per pair). It can be seen that the features cover densely the model area, without an obvious preference for the face elements. We attribute this to the need of the model to inspect the background as much as the face elements in order to decide for the presence of a face or not. Feature pairs also show no clear pattern, suggesting that random proposals might be a reasonable option during training (we currently use exhaustive greedy search, root node first, leave nodes given the root node).

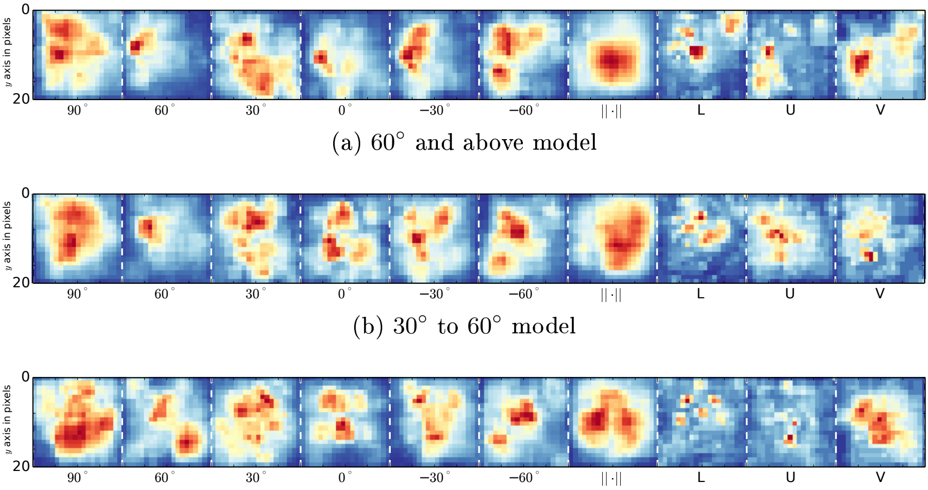
References

1. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge (2008)
2. Dollár, P., Tu, Z., Perona, P., Belongie, S.: Integral channel features. In: BMVC. (2009)
3. Koestinger, M., Wohlhart, P., Roth, P.M., Bischof, H.: Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In: ICCV BeFIT workshop. (2011)



(a) 60° and above model (b) 30° to 60° model (c) Frontal model

Figure 6: Components of the learned model (and example training sample). Red indicates areas with higher influence for the score decision.

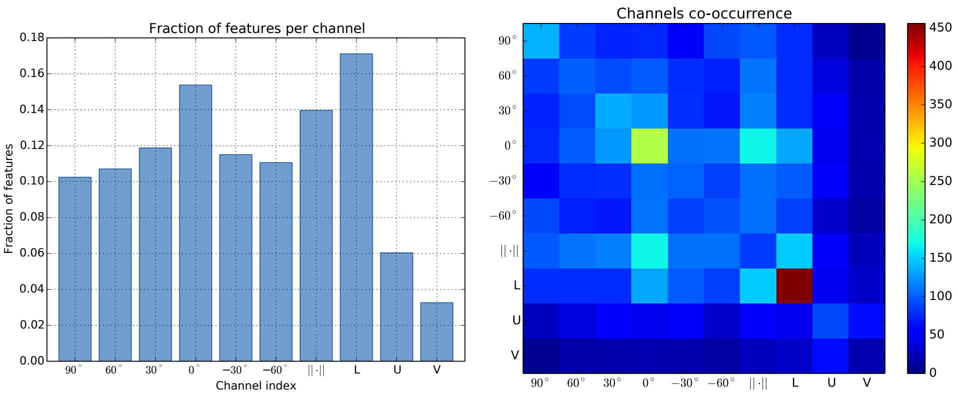


(a) 60° and above model

(b) 30° to 60° model

(c) Frontal model

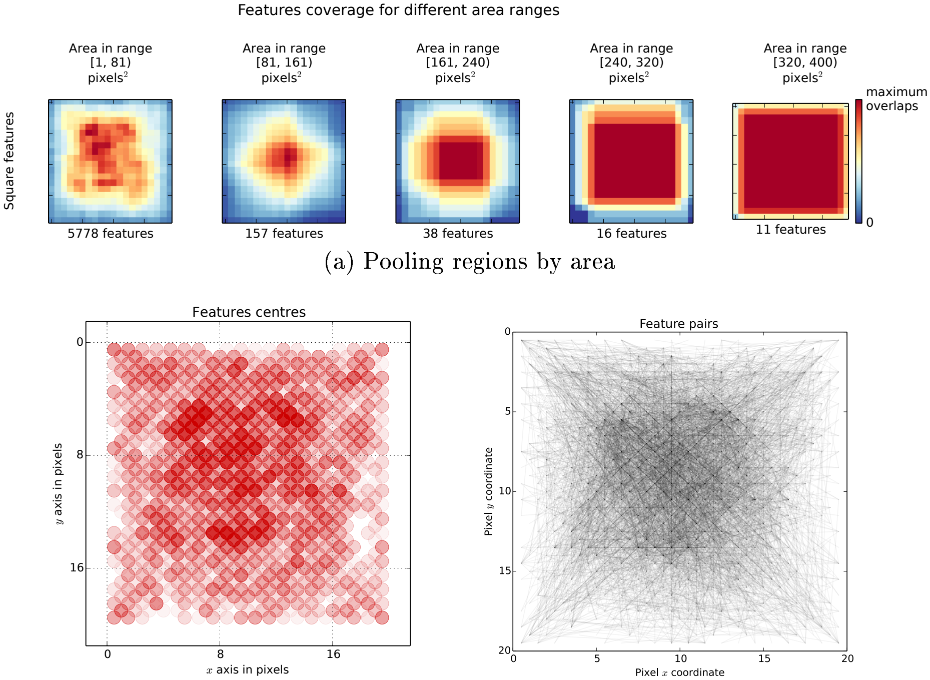
Figure 7: Per channel view of the pooling regions. Red indicates areas with higher influence for the score decision, colour code normalized per channel.



(a) Frequency of channel usage

(b) Channels co-occurrence

Figure 8: Statistics of the channels usage for the frontal model. Other models show a similar pattern.



(a) Pooling regions by area

(b) Centre of the pooling features. Features are well distributed all around the template.

(c) Feature pairs. No clear pattern is visible, pairs are spread all over the template.

Figure 9: Statistics of the channel usage for the frontal model, other views show a similar pattern.